

REMARKS

Status of Claims

Claims 1-15 are pending in this application. By this Amendment, claims 7, 10-11, 14-15 have been amended and new claims 16-20 have been added. Reconsideration is respectfully requested in view of the above Amendments and the following remarks.

Interview Summary

Applicants thank examiner Kiss for granting the interview on September 8, 2005 and for considering the arguments regarding the deficiencies of the prior art, including Fedrov et al. During the interview, the examiner broadly construed a wizard application to include a browser application. There was no agreement reached regarding the breadth of the examiner construction of the term “wizard” as used within the application and the claims. The prior art fails to disclose, among other things, a user interface that integrates a web component in a host wizard by utilizing an extension interface.

Rejections under 35 U.S.C. §102(b)

Claims 7, 10, and 11 were rejected under 35 U.S.C. § 102(b) as being anticipated by Alex Fedorov, *et al.* “Professional Active Service Pages 2.0,” 1998, Wrox Press Ltd. (hereinafter “Fedorov *et al.*”). This rejection is respectfully traversed.

“A claim is anticipated only if each and every element as set forth in the claim is found, either expressly or inherently described, in a single prior art reference.” *Verdegaal Bros. v. Union Oil Co. of California*, 814 F.2d 628, 631, 2 USPQ2d 1051, 1053 (Fed. Cir. 1987).

With respect to independent claim 7, Fedorov fails to disclose “providing a host wizard that defines an extension interface to respond to navigation events; providing a web component comprising: a web page, said web page containing a wizard control area and a

control interface, and utilizing the extension interface to perform recursive navigation between said web component and said host wizard.” The office action has referenced Fedorov at pages 423-426, which discloses:

“Over the past several years users have becoming accustomed to dialog box based wizards . . . it is now possible to create . . . rich user interfaces . . . web-based dialog boxes . . . The dialog box user-interface is provided by equakeget.htm and the server-side calculations are provided by seismic.asp.”

Fedorov discloses a web-based dialog, wizard, that is used to collect information and produce calculations. Fedorov does not disclose the use of a host wizard. The Seismic.asp and equakeget.htm, disclosed by Fedorov, represent two components of a web-based calculator wizard. The Office Action contends that the browser application anticipates the claimed host wizard and the navigation to the host wizard. The browser application provides the environment for retrieving and displaying the web component. The browser application is not “a wizard” as defined by applicant’s specification page 2, lines 7-22. Accordingly, Fedorov discloses one wizard that functions as a calculator and, the wizard has its functionality determined by a web file and a script file. However, Fedorov fails to disclose “a host wizard,” and “the **integration** of the “host wizard and web component.” Accordingly, for at least the reasons set forth above, claim 7 is allowable over Fedorov.

Furthermore, Fedorov fails to disclose a host wizard that defines an extension interface that responds to navigation events. The host wizard defines the extension interface to allow navigation events triggered by the web component to retrieve the appropriate host pages.

Independent claims 10 and 11 recite limitations similar to those of independent claim 7. Therefore, independent claims 10 and 11 are allowable for at least the reasons set forth with respect to independent claim 7.

Rejections under 35 U.S.C. §102(e)

Claims 1-6, 8, 9, and 12-15 were rejected under 35 U.S.C. § 102(e) as being anticipated by U.S. patent No. 6,574,791 to Gauthier *et al.* This rejection is respectfully traversed.

With respect to independent claim 1, Gauthier fails to disclose “invoking said one or more sub-wizard components during said host-wizard component execution; and transferring control from said host-wizard to said one or more sub-wizard components.” The office action has referenced column 9, line 56-column 10, line 5, as anticipating the limitations of this claim. The section cited by the office action discloses:

“The WizardManager class defines objects which control the execution of multiple subwizards within the target wizards. This class also references the GUI elements making up the overall target wizard. This class preferably includes constructor methods for creating a new WizardManagerFrame object, a new WizardManagerButtonPanel object, a new WizardManagerSelectionPanel object, and a new WizardManagerLogoPanel object, and constructor methods that creating these objects from specified existing meta data. These constructor methods would call corresponding constructor methods on the corresponding class. This class also preferably includes an AddWizard() method for adding new subwizards to the overall wizard skeleton. This class also preferably includes methods for launching the overall wizard, and launching selected subwizards. The class also preferably includes a describe content method used to export meta data from the class objects.”

In Gauthier, a “WizardManager” controls the execution of the subwizards which define the Target wizard, col 13, lines 25-35. The execution of the subwizards is centralized and coordinated by the WizardManager. The Gauthier disclosure differs from the invention defined by claim 1 because the control functions of Gauthier are centralized and not distributed, in Gauthier, as required by Claim 1. The Office Action further contends that column 14, line 9 through column 15, line 24 discloses the claimed “transferring of control to the subwizard. Gauthier discloses WizardState objects that are utilized to generate code that implement the wizard functions. Nothing in the cited section discloses control is passed to

the subwizard during the execution of the host wizard. Gauthier does not define a host wizard. Unlike Gauthier, claim 1 requires a host-wizard to transfer control directly to a sub-wizard, and the sub-wizard controls its own execution. Therefore, Gauthier fails to disclose a transfer of control to one or more sub-wizard components. Accordingly, for at least the reasons set forth above, claim 1 is allowable over Gauthier.

Claim 2 depends from claim 1 and further defines novel features of the claimed invention. Accordingly, claim 2 is allowable by virtue of its dependence on claim 1. Additionally, claim 2 is allowable because Gauthier fails to disclose "wherein one or more sub-wizard components are browser based object components." The office action has referenced column 18, line 66-column 19, line 30, as anticipating this limitation. The section cited by the office action discloses:

"Turning now to FIG. 6, the preferred embodiment implementation of the WizardMetaDataManager 128 is illustrated in more detail. As discussed above, the WizardMetaDataManager 128 is used to persist and retrieve target wizard meta data. In the preferred embodiment, the WizardMetaDataManager 128 implementation includes a WizardMetaDataManager object, a WizardMetaData object, a WizardMetaDataFormDescription object, a WizardMetaDataPanel object, and a WizardMetaDataFrame object.

The WizardMetaDataManager object preferably includes a ReuseExistingMetaData() method and a SaveWizardMetaData(). The SaveWizardMetaData() displays the WizardMetaDataPanel object which prompts the developer to select the meta data to be stored and specify a wizard meta data file in a form specified by the WizardMetaFormDescription object. The ReuseExistingMetaData() method displays the WizardMetaDataPanel and prompts the user to select a set of wizard meta data. The method retrieves the wizard meta data and walks through it to recreate an **internal** set of wizard framework objects.

The WizardMetaData object includes the actual meta data for the target wizard. This would preferably include all of the components of the target wizard and their interrelationships. The storage would preferably be done in a **language neutral format** to facilitate ease of restoring. The WizardMetaDataFormDescription object describes the format used for storing the wizard meta data. For example, the wizard meta[l] data could be described using extensible markup language (XML) and an accompanying wizard framework specific document content description. The WizardMetaDataPanel object and a WizardMetaDataFrame object provide the GUI interface to the developer."

Gauthier teaches a WizardMetaDataManager is able to retrieve stored meta data, the meta data preferably being stored in a language neutral format. Gauthier further discloses that it is possible to describe the meta data using XML and a specific document content description. XML is an example of the language neutral format that aids in defining the meta data. Gauthier further discloses that each subwizard includes a Wizard object, a WizardState object, a WizardDefault object, a WizardStateController object, WizardPanel object(s), and a WizardCodeGenerator object, column 14, lines 9-12. However, Gauthier fails to disclose sub-wizards being defined to include browser based object components.

Furthermore, the Office Action further contends that because XML may be used to define web objects that Gauthier must utilize XML in this fashion. The Office Action is attempting to broaden the disclosure of Gauthier by referring to a document whose date of publication is after the filing date of the current application. This is an improper rejection under 35 U.S.C. §§ 102 and 103. To be a proper rejection the document cited must be prior art, that is, publicly disclosed before the effective filing date of applicant's patent application. The document cited, current XML specification, is not prior art. Therefore, for at least the reasons set above, claim 2 is allowable over Gauthier.

Claim 3 depends from claim 1 and further defines novel features of the claimed invention. Accordingly, claim 3 is allowable by virtue of its dependence on claim 1. Additionally, claim 3 is allowable because Gauthier fails to disclose "wherein on or more sub-wizard components are operating system based application component object extensions." The Office Action has referenced column 6, line 42-46. The section cited by the office action discloses:

"The operating system 122 provides the basic functionality that controls the computer system 100. Operating system 122 can comprise any suitable operating system, such

as IBM's AS/400, OS/2, Microsoft's Windows, Java and the various flavors of UNIX”

In this section, Gauthier teaches the use of an operating system and the different types of operating systems. Gauthier, in column 7, lines 45-58 further discloses a wizard framework that defines the basic elements of a wizard. “This framework defines the core functions of the solution, those elements that are required and cannot be extended by developer. The framework also defines extensible functions of the solution, those that can be customized and extended by the developer. The customization/extension quality of framework mechanisms is extremely valuable because the cost of customizing or extending a framework is much less than the cost of a replacing or reworking an existing solution.” Although Gauthier refers to extensions of particular functions, these types of extensions are limited specifically to objects within the wizard framework. There is no teaching of an extension of any operating system based application components. Applicant’s specification at page 14-15 attempts to briefly illustrate what an operating system based extension comprises. Therefore, for at least the reasons set forth above, claim 3 is allowable over Gauthier.

With respect to independent claim 4, Gauthier fails to disclose “a host-wizard, said host-wizard having an interface adapted to communicate with other wizards and a host-wizard navigational component adapted to transfer control to other wizards.” The office action has referenced Gauthier at column 9, line 55 – column 10, line 5, column 10, lines 57-67 and column 14, lines 9-14 which discloses:

“The Wizard class provides objects to list panels in subwizard and their associated panel flow order. The Wizard class preferably defines constructor methods that call to create new WizardState objects, new WizardStateController objects, new WizardDefault objects and new WizardCodeGenerator objects for each new subwizard when called by the WizardWizard. The Wizard class also preferably includes methods to add new panels to a subwizard and set panel flow when called by

the WizardWizard and WizardDesigner. [T]he class also preferably includes a describe content method used to export meta data from the class objects.”

“Each subwizard in the Target wizard 129 preferably includes a Wizard object, a WizardState object, a WizardDefault object, a WizardStateController object, WizardPanel object(s), and a WizardCodeGenerator object. The Wizard object includes a list of panels that comprise the subwizard, and the corresponding panel flow order.”

As stated above, the “WizardManager” centralizes the control and the execution of the subwizards which define the Target wizard. Therefore, Gauthier does not disclose **a host navigational component** adapted to transfer control to other wizards. Moreover, Gauthier fails to disclose “one or more sub-wizard components, said one or more sub-wizard components having sub-wizard interfaces adapted to communicate with other wizards **and sub-wizard navigational components** adapted to transfer control to other wizards”. The office action further argues that column 10, lines 30-33 allows for selection and invoking of the subwizard. Gauthier does not disclose that selecting the subwizard in the manager selection component invokes the subwizard. Accordingly, for at least the foregoing reasons, claim 4 is allowable over Gauthier.

Claims 5 and 6 depend from claim 4 and further define novel features of the claimed invention. Accordingly, claims 5 and 6 are allowable by virtue of their dependence on claim 4. Additionally, claims 5 and 6 are allowable for at least the reasons set forth above with respect to claims 2 and 3.

Independent claims 8 and 9 recite similar limitations to that of independent claim 1. Therefore, independent claims 8 and 9 are allowable for at least the reasons set forth above with respect to independent claim 1.

With respect to independent claim 12, Gauthier fails to disclose “providing at least one navigation component on each of said first and second wizards, said navigation

components allowing sequential progression or regression through said first and second wizards to chain said second wizard to said first wizard.” The office action has referenced Gauthier at column 9, line 55 – column 10, line 5, column 10, lines 57-67 and column 14, lines 9-14, reproduced above.

Gauthier at column 10, lines 20-30 discloses the WizardManagerButtonPanel class provides a panel with a plurality of buttons for use on the target wizard interface. These buttons would typically include standard GUI interface buttons, such as BACK, NEXT, FINISH, CANCEL and HELP. However, the buttons are not on **each wizard**, the buttons are centralized using a WizardManagerFrame, column 10, line 5-40. Furthermore, as noted above the execution and control of the subwizards of Gauthier are centralized; so, the subwizards are not chained through the use of navigational components. The Office Action further contends that panel flows is admitted prior art. However, the panel flows do not describe how to integrate separate and distinct wizards. Accordingly, for at least the foregoing reasons, independent claim 12 is allowable over Gauthier.

Claim 13 depends from claim 12 and further defines novel features of the claimed invention. Accordingly, claim 13 is allowable by virtue of its dependence on claim 12. Additionally, claim 13 is allowable for at least the reasons set forth above with respect to claims 2 and 3.

Independent claims 14 and 15 recite similar limitations to that of independent claim 12. Therefore, independent claims 14 and 15 are allowable for at least the reasons set forth with respect to independent claim 12.

CONCLUSION

Having demonstrated that the cited references fail to disclose or suggest the invention as claimed, this application is in condition for allowance. Accordingly,

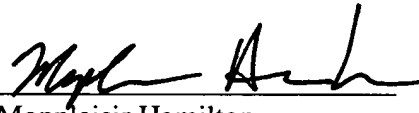
applicant requests early and favorable reconsideration in the form of a Notice of Allowance.

If there are any questions regarding this amendment or the application in general, a telephone call to the undersigned would be appreciated, since this should expedite the prosecution of the application for all concerned.

If necessary to effect a timely response, this paper should be considered as a Petition for an Extension of Time sufficient to effect a timely response. Please charge any deficiency in fees or credit any overpayments to Deposit Account No. 19-2112 (Attorney Docket No. MFCP.88142).

Respectfully submitted,

Date: October 19, 2005


Monplaisir Hamilton
Reg. No. 54,851

SHOOK, HARDY & BACON L.L.P.
One Kansas City Place
2555 Grand Boulevard
Kansas City, Missouri 64108-2613
Phone: (816) 474-6550
Facsimile (816) 421-5547